(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 0 978 968 B1

(12) **EUROPEAN PATENT SPECIFICATION**

(45) Date of publication and mention of the grant of the patent:
23.02.2005 Bulletin 2005/08

(51) Int Cl.[7]: **H04L 12/56**, H04Q 11/04

(21) Application number: **99250262.5**

(22) Date of filing: **03.08.1999**

(54) **High speed cross point switch routing circuit with flow control**

Leitweglenkungschaltung einer Schaltmatrix mit hoher Geschwindigkeit und mit Flu kontrolle

Circuit d'acheminement d'une matrice de commutation à grande vitesse avec réglage de débit

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE

(30) Priority: **05.08.1998 US 129662**

(43) Date of publication of application:
09.02.2000 Bulletin 2000/06

(73) Proprietor: **Vitesse Semiconductor Corporation**
Camarillo, California 93012 (US)

(72) Inventors:
- **Mullaney, John P.**
  Minneapolis, Minnesota 55436 (US)
- **Lee, Gary M.**
  Camarillo, California 93012 (US)

(74) Representative:
Müller, Wolfram Hubertus, Dipl.-Phys. et al
Patentanwälte
Maikowski & Ninnemann,
Postfach 15 09 20
10671 Berlin (DE)

(56) References cited:
WO-A-95/30318
WO-A-96/42158
US-A- 4 958 341

EP 0 978 968 B1

Description

FIELD OF THE INVENTION

[0001] The present invention relates to high speed data communication network systems and, more particularly, to a bit and word synchronized high speed serial switch routing system.

BACKGROUND OF THE INVENTION

[0002] Modern, high-speed data communication and transmission frequently involves the use of multiple transmitters and receivers communicating with one another, or with multiple memory devices, for example, over high-speed data transmission lines. Such high-speed data transmission generally imposes stringent requirements on clock synchronization. Further, high speed data communication systems require large amounts of data to be sent to various different locations or devices comprising a communication network. This is typically performed by using networking devices, conventionally termed switches or routers, which receive data from a particular transmitter and reconfigure a signal path in order to send the data to a designated recipient. Conventional switches or routers implement a "switch fabric" using integrated circuits to provide a data route from a receiving input port to a correct output port of the networking device (the switch). The data routes often implemented in high-speed switch fabrics are generally one bit wide. Thus, for such switches the switch fabric routes data over a plurality of serial data paths.

[0003] Modern high-speed communication systems place inordinate demands upon the performance requirements of the switch of a network switching system. The switch must be able to operate at a sufficiently high bandwidth such that signal processing is not unduly delayed while data is being transferred. Further, connections are frequently being made and broken, such that delays often occur while waiting for a connection. In addition, the various possible routes through the switch fabric from one port to another port are not always of equal length. Therefore, signal path lengths, and signal delays, change with each reconfiguration of the switch.

[0004] FIG. 1 illustrates a prior art semi-schematic simplified block diagram of a network switching system as is e.g. disclosed in US 4 958 341. As illustrated in FIG. 1 a crosspoint switch or router circuit is typically implemented as a number of integrated circuit components configured on a printed circuit board or card 10. The switch need not be a crosspoint switch, delta or other switch types may also be used. The switch 10 comprises a switch matrix or fabric 12 which is reconfigurable under control of a central processing unit 14 to receive data from a switch port circuit 16 and route the data to a designated recipient switch port.

[0005] Data is transmitted to, and received from corresponding switch ports by a multiplicity of transceiver circuits 18. The transceiver circuits are configured to move data to and from a particular user application through transmit and receive FIFOs over parallel interface busses. Parallel data is serialized and directed to a particular switch port over a high-speed serial interface. Likewise, serial data is received by the transceiver 18 from a corresponding switch port 16. The transceiver deserializes the serial data and interfaces with a user application circuit through a receive FIFO over a parallel data bus.

[0006] Each of the transceivers 18 typically include a clock and data recovery circuit (CDR) 20. The CDR locks onto the incoming serial data stream in order to recover clock information suitable for controlling the timing of the various registers comprising the transceiver. As noted above, when control signals to the crosspoint switch change the switch configuration, the delay through the crosspoint also changes. Because of this delay change, the CDR must realign itself to the phase of the new data stream.

[0007] In addition, prior art-type transceiver circuits are typically constructed with their own reference clock generator 22. The reference clock generator functions as a frequency reference for the CDR 20 such that the CDR 20 is able to operate in "fly wheel" mode during periods when there is no data. Since reference clock generators may be frequency mismatched by approximately 100 PPM with respect to one another, it is possible that a serial bit stream developed by one transceiver and received by a second transceiver is sufficiently shifted in phase such that a certain number of bits might be lost in each transmitted frame. Moreover, during long periods of transmission from one transceiver to another data may be lost due to timing drift because the clocks of various transceivers may be of slightly different frequencies. This necessitates periodic switch reconfiguration to force transceiver resynchronization, with prior art switches usually having a cell period defining a maximum continuous transmission length. Each time transmission is interrupted to force resynchronization, of course, effective switch bandwidth is reduced. Further, variations due to transceiver frequency mismatch and the changing delay paths through a crosspoint matrix are random in direction as well as frequency. Adjusting a CDR in response to a serial data stream received from a first transceiver may result in over-correction, particularly if the serial data stream from a next transceiver is jittered in the other direction.

[0008] In addition, the crosspoint switch delay change caused by reconfiguration can be larger than one bit time, such that word or frame realignment must also be performed by the receiving transceiver. Word or frame realignment is a generally lengthy process requiring many bit times to perform. Thus, a dead period is induced in the data stream which contains no valid data. In asynchronous transmission mode switches, for example, this realignment dead period reduces the effective bandwidth of a network switching system by approx-

imately 10-20%. Moreover, phase recovery circuitry must be made as fast as possible to compensate for transceiver frequency mismatch and to minimize realignment induced dead time. Conventional systems typically use up an additional 10-20% of bandwidth in order to provide a minimum number of transitions to guarantee that the serial data stream comprises a sufficiently high transition rate to support fast phase recovery circuits.

[0009] Serial data transmission may also be synchronous. In synchronous data transmission the sequence of binary "ones" and "zeros", making up the data stream, occurs with reference to a data bit cell defined by a uniform or single-frequency clock signal transmitted with the data. Transmitting the clock signal together with the data, however takes up valuable bandwidth, increases high speed line requirements, and reduces the data transmission capability of the system. In addition, word alignment must still be performed.

[0010] The effects of jitter, or bit shift, in a serial data stream are illustrated in FIG. 2. Data has been phase-locked to a bit clock signal wherein data is stable within a particular bit period such that it may be strobed into an input register on the falling edge of bit clock. Given perfect phase and frequency lock, the periodicity of the bit clock signal might serve to define synchronous bit cells; a logical high data occurring within a code bit cell representing a logic ONE, a logical zero on data occurring within a code bit cell representing a logic zero. The data sequence illustrated would therefore be read as 11011000.

[0011] Phase jitter, frequency mismatch, andior a delay change through the switch matrix, has displaced, or shifted, the serial data stream by approximately 90 degrees in phase. Data stability, of the late data stream, occurs outside of the intended code bit cell, and into the next code bit cell, causing the data stream sequence to be incorrectly read as 01101100 rather than 11011000. Thus, it can be seen that by merely shifting a particular serial data stream by approximately 90 degrees in phase, the binary sequence comprising a data word, as represented within a frame defined by a word clock signal, causes the word to lose all meaning.

[0012] The random nature of data shift can be appreciated by referring to FIG. 3. Shifts in the nominal position of a data transition edge due to timing fluctuations result in a normal distribution of possible transition edges distributed with respect to time around the occurrence of the bit clock timing edge. If the bit clock period is used to define the code bit cell boundaries, there would be an approximately 50% probability that a transition edge, representing a transition from 1 to 0, or 0 to 1, would be shifted early or late and therefore not captured in the proper code bit cell, giving rise to a data word error. A code bit cell should properly have its boundaries symmetric about the mean of an expected data value. However, because of a multiplicity of reference clock signals provided in prior art-type transmission systems, bit cell boundaries must be inordinately wide in order to accommodate the expected transition edge distribution pattern. Widening the bit period necessarily requires that a system bandwidth be consequently reduced, reflecting a loss of transmission capability. Accordingly, some other means must be provided to ensure that all of the component elements of a multiport transmission system be at least frequency locked together, such that only phase recovery is necessary to correctly place the transition edges of a serial data stream within an appropriate code cell boundary.

[0013] The same reasoning holds true for word synchronizing a 2.125 GHz serial data stream. A word detection window (word clock) must be able to accommodate variations in its own frequency and phase in order to provide for accurate detection and capture of a data word from serial data running at slightly variable channel rates. If a word clock signal were to be bit-shifted by the same approximately 90 degrees in phase from the bit clock signal, the same type of data read error would occur as if the bit clock signal were shifted. Thus, it will be understood that in addition to having each of its component elements frequency-locked together, an effective high-speed data transmission system must also provide for a word clock signal which is frequency-locked to a bit clock. Moreover, the word clock signal must accurately define the beginning of a data word and, thus, must be consistent across all of the component elements of such a transmission system.

[0014] In addition, many prior art switches utilize a processing unit to determine a switch configuration and provide flow control signals for controlling the flow of information through the switch. The processing unit receives connection requests and transceiver status signals over a common data bus accessed by all transceivers connected to the switch. Thus, at any given time, transceivers are requesting access to the common data bus to place connection requests and to provide transceiver status signals, such as signals indicating that the transceiver is unable to accept additional data due to the transceivers input buffer being full.

[0015] The use of a common data bus and processor receiving information from a plurality of transceivers over the data bus may result in delays in data communications. That is, the data bus may not be accessible at any given instance due to the data bus already being in use by another transceiver. Accordingly, the system design must take into account delays due to use of a common data bus in determining when to transmit transceiver input buffer full status, and other signals, and additionally switch connections may be delayed due to delays in providing the processing unit the connection requests. Thus, the use of a processing unit and common data bus further decreases the bandwidth of the switch.

SUMMARY OF THE INVENTION

[0016] The present invention therefore provides a

high speed network switching apparatus, comprising a switch including a plurality of switch ports, each switch port defining a transmission channel and adapted to transmit and receive a high-speed serial data stream. Further a switch fabric coupled to the plurality of switch ports, the switch fabric routing data among and between the switch ports and a plurality of transceiver circuits, each transceiver circuit configured to transmit and receive a high speed serial data stream between a corresponding one of the plurality of switch ports so as to establish a transmission channel between a corresponding transmitting transceiver circuit and a corresponding receiving transceiver circuit. The data stream includes command and data words comprising a data portion and a header portion, wherein the header portion includes overhead bits configured to provide a ready-to-receive indication from a receiving transceiver circuit to a transmitting transceiver circuit when the overhead bits are in a first binary sequence, a not-ready-to-receive indication from a receiving transceiver to a transmitting transceiver when the overhead bits are in a second binary sequence, the switch adaptively routing the overhead bits from the corresponding receiving transceiver circuit to the corresponding transmitting receiver circuit.

DESCRIPTION OF THE DRAWINGS

[0017] These and other features, aspects and advantages of the present invention will be more fully understood when considered with regard to the following detailed description, appended claims and accompanying drawings wherein:

FIG. 1 is a semi-schematic block diagram of a prior art crosspoint switch and transceiver;

FIG. 2 illustrates a series of waveform diagrams showing the effects of timing error on a serial data stream;

FIG. 3 illustrates the random nature of bit displacement and its effects on word alignment;

FIG. 4 is a semi-schematic block diagram of an example of a multiplicity of high speed serial transceiver ports coupled to a cross-switch routing circuit of the present invention;

FIG. 5a is a semi-schematic block diagram of the transceiver circuit of FIG. 4 incorporating circuitry for bit and word synchronization;

FIG. 5b is a semi-schematic block diagram of the switch circuit of FIG. 4 incorporating circuitry for developing a global clock domain for bit and word synchronization;

FIG. 6 is a flow diagram illustrating a word synchronization process of the present invention;

FIG. 7 illustrates command and data word formats for illustrating 34-bit transmission characters including two overhead bits for providing self-routing and low level flow control of the present invention; and

FIG. 8 is a semi-schematic block level diagram illustrating the use of overhead bits to accomplish low level flow control in a multi-transceiver implementation of the present invention.

DETAILED DESCRIPTION

[0018] FIG. 4 illustrates a semi-schematic block diagram of a packet-based switching system of the present invention. The system includes a 16 X 16 synchronous serial crosspoint switch circuit 50. Up to 16 high speed serial transceiver port cards 52 (only four of which are shown) are connected to the crosspoint switch circuit by pairs of suitable transmission lines 57a, b. In the embodiment described the crosspoint switch circuit operates at an aggregate bandwidth of up to approximately 32Gb/s. The crosspoint switch circuit unit 50 includes a conventional 16 X 16 crosspoint switch fabric 53 configured to provide sixteen connections from one side of the switch fabric to another side of the switch fabric on a selective basis. The fabric 53 is connected to receive serial data from, and send serial data to, 16 bi-directional switch ports, commonly indicated at 54. Each of the switch ports are in turn connected to transmit and receive serial data from a corresponding one of the port cards 52 over the transmission lines at a serial data rate of about 2.125 GHz. For purposes of clarity and for ease of explanation, only four of bi-directional switch ports 54 are shown in the example of FIG. 4, but those having skill in the art will recognize how to expand the representations of both the switch ports 54 and the transceiver port cards 52 so as to accommodate a 16 X 16 switch fabric. Implementations of crosspoint switch matrixes or fabrics are well known in the art, and therefore the switch matrix or fabric 53 crosspoint switch (or cross-switch router) requires no further elaboration herein. It is sufficient to mention that the switch matrix or fabric is a 16 X 16 matrix which receives incoming serial data from a selected one of the port cards 52 and routes the serial data to an appropriate recipient port card, designated as the addressee in a data packet header at the incoming serial data. Further, other switches, such as a Delta switch, may be used in place of the crosspoint switch of the example of FIG. 4.

[0019] An arbitration logic and switch control circuit 55 determines the configuration of the crosspoint switch fabric 53. The switch control circuit communicates with logic circuitry 56 incorporated into each of the switch ports 54 in order to ensure that the data received from the transmitter is directed to a correct designated recipient. The switch control circuit implements a round robin arbitration scheme for allocating switch connection requests. Circuits for implementing round robin arbitration schemes are known in the art. Alternatively, the arbitration logic and switch control circuit could maintain a record of all switching and routing transactions in a port connection table, and thereby identify sender/recipient pairs and keep track of available connections through the switch fabric.

[0020] As will be developed more fully below, routing, or connection, requests (CRQs) are made by a transmitting port to the arbitration logic and switch control circuit 55, which appropriately configures the matrix 53. The logic circuity 56 additionally provides the switch control circuit connection requests overhead bits to effect flow control of data transmitted through the switch.

[0021] A global, system wide clock signal is provided on the switch circuit unit 50 and defines a global word clock (WCLK) signal, which is a 62.5 MHz signal in the described example. A synchronized bit clock timing signal is developed through a CMU circuit 58 using the word clock signal, with the bit clock timing signal being a 2.125 GHz signal in the described example. The WCLK signal is provided by an external 62.5 MHz crystal oscillator, which is coupled to the switch circuit in conventional fashion, but some other suitable reference clock generation circuit may also be used. The bit clock timing signal is directed, globally, to each of the bi-directional switch ports 54 comprising the switch circuit unit 50. Defining the bit clock timing signal for each of the switch ports from a single input reference clock signal (WCLK) has important implications to the synchronous bi-directional data transmission characteristics of the system. Since each of the switch ports operate off of a unitary timing signal developed from a single timing reference, it will be understood that each of the switch ports 54 will operate in a synchronous, albeit possibly phase shifted, fashion with the others.

[0022] Each switch port 54 comprises a receiver section including a serial-to-parallel data converter 60 (also referred to as a deserializer or DMUX). The DMUX is configured to receive a serial data stream transmission from the transmitter section of a corresponding transceiver port card 52 and convert the serial data into a parallel data word (referred to herein as a transmission character). In the example described, the serial-to-parallel converter 60 receives incoming data transmissions at a 2.125 Gb/s data rate and outputs a 34-bit transmission character comprising a 32-bit data word, plus two overhead bits, at a parallel data rate of about 62.5 MHz.

[0023] Similarly, each switch port 54 comprises a transmitter section including parallel-to-serial data converter 62 (also referred to as a serializer or MUX) which performs a similar function to the serializer 60, but in reverse. The parallel-to-serial converter 62 receives a 34-bit transmission character (a 32-bit data word, plus two overhead bits) which has been routed to the corresponding switch port through the switch fabric 53 at an input parallel data rate of approximately 62.5 MHz. The parallel-to-serial converter converts the parallel data into a serial data stream suitable for transmission to a receiver portion of the port card 52 at a serial data rate of approximately 2.125 Gb/s. Thus, the 62.5 MHz WCLK signal, distributed by the CMU clock circuit 58, is used as a master strobe to clock 34-bit parallel data out of the serial-to-parallel converter 60 to the port logic circuit 56. The WCLK signal is also used to clock 34-bit parallel data from the port logic circuit 56 into the parallel to serial converter 62. All timing signals, whether serial bit timing signals or parallel word timing signals, used by each of the switch ports, therefore, is developed by the CMU clock circuit 58 in response to the system wide reference WCLK.

[0024] Each of the transceiver port cards 52 are typically constructed to include a mix or combination of transceiver circuitry and circuitry related to a particular user's application. In a typical configuration, a transceiver port card includes physical layer circuitry 61, 63 for a given communication protocol, and data buffer circuitry that manages the information flow and formatting between downstream user application circuitry and the transceiver. In the example of FIG. 4, the data buffer circuitry comprises transmit and receive FIFOs 64 and 66, respectively. The transmit and receive FIFOs are each coupled to the transceiver circuitry over a parallel data interface. In the example described, data is clocked to the transceiver at the 62.5 Mb/s parallel data rate.

[0025] In many communications applications, however, the parallel data interface to the transmit and receive FIFOs will operate at a different frequency than the 62.5 MHz word clock used by the switch card and the transceiver port card's transmit and receive circuitry. In this case, the transmit and receive FIFOs 64 and 66 are implemented as synchronous, dual port FIFOs, whose dual clock ports are used to elastically span any discontinuous clock boundaries between the transmission side and the media side. In addition, the transmit and receive FIFOs are made large enough to function as data queues or data buffers for each port card's transceiver circuitry. The transmit and receive FIFOs may be implemented as register stacks, string buffers, and the like, but are preferably implemented as dual-port, parallel data buffer, integrated circuit memory elements. Suitable transmit and receive FIFOs 64 and 66 include FIFO devices able to operate at speeds up to 67 MHz. Such a FIFO is exemplified by the IDT7236 series of synchronous FIFOs, manufactured and sold by Integrated Device Technology, among others.

[0026] Each transceiver port circuity 68 comprises a transmitter section, indicated as TX, and a receiver section, indicated as RX. The transmitter section includes a serializer (not shown in the example of FIG. 4) for converting 62.5 Mb/s parallel data from the transmit FIFO 64 into a serial data stream suitable for transmission to the receiver section of a corresponding switch port. Likewise, the receiver section incorporates a serial-to-parallel converter (also not shown) for converting a high speed serial data stream, from the transmitter section of the corresponding switch port, into 62.5 MHz parallel data suitable for receipt and storage by the receive FIFO 66. The serial interconnect between a transceiver port circuitry 68 and the corresponding switch port of switch circuit card 60 is a bi-directional serial interface and operates to transmit and receive high-speed, serial data signals at 2.125 Gb/s. In a manner that will be described

in greater detail below, the serial data streams communicated between a transceiver port and a switch port, and the parallel data communicated between a transceiver port circuitry 68 and its associated FIFOs 64 and 66, are bit and word synchronized to the bit and word timing signals used to strobe serial and parallel data by the timing elements of the switch card 50.

[0027] The transceiver bit clock (i.e., the timing signals used to define the bit or cell period of serial data) is recovered from an incoming serial data stream provided to the transceiver port's receiver section by the switch port 54. Recovered bit clock strobe transitions are also used to clock an outgoing serial data stream from the transmitter to a corresponding switch port 54. Accordingly, the serial data stream provided to the switch port 54 by the transceiver 68 will be at the same frequency as the bit clock of the switch port, and need only have its phase evaluated in order to ensure proper bit phase alignment because the switch card 50 defines the timing parameters of a serial data stream directed to the transceiver port circuitry 68.

[0028] Referring now to FIGS. 5a and 5b, there is shown a semi-schematic block diagram of an embodiment of a transceiver port circuitry 68 coupled to a corresponding switch port 54 of a switch card over a bi-directional 2.125 Gb/s differential serial data link. The transceiver circuit is illustrated in FIG. 5a, while the corresponding switch port and switch unit circuitry is illustrated in FIG. 5b. As is the case with the example illustrated in FIG. 4, the transceiver port circuitry 68 is connected to transmit and receive FIFOs, 64 and 66 respectively, over 62.5 MHz parallel data interface busses adapted to communicate parallel data between the transceiver port circuitry 68 and the FIFOs. The data interface connection from the transmit FIFO 64 comprises a 32-bit parallel transmission data bus TXIN[31:0], which is clocked into the transceiver on a transition edge of the transceiver master word clock timing signal WCLK. As will be later discussed, the transceiver develops two word clocks, a transceiver transmit word clock and a transceiver receive word clock. In order to increase the ease of interfacing with the transceiver, the transceiver transmit word clock is designated as the transceiver master word clock.

[0029] A 2-bit transmission type signal TXTYP[1:0] is also provided on the parallel interface and, if the transceiver is put into a first configuration, defines the type of data word being transmitted. On the other hand, as will be described below, if the transceiver is put into a second configuration, the TXTYP[1:0] signals directly control the configuration of two overhead bits appended to a command or data word in the MSB positions. The two overhead bits function to define flow control signals in the second configuration.

[0030] The data interface connection between the transceiver and the receive FIFO 66 is a 32-bit parallel receive data bus RXOUT[31:0]. The receive data bus is clocked out of the transceiver on a transition edge of the transceiver master WCLK timing signal. A two bit receive word type signal RXTYP[1:0] is also provided on the parallel interface. The receive word type signal defines the type of data word being received if the transceiver is put into the first configuration. If the transceiver is put into a second configuration, however, the RXTYP [1:0] signal reflects the configuration of two overhead bits appended to a command or data word and received over the serial channel of the transceiver.

[0031] In addition to these parallel signal busses, both the transmit FIFO 64 and receive FIFO 66 host numerous command and control signals, several of which are coupled to the transceiver circuit and several of which are coupled to application control circuitry (63 of FIG. 4). Those having skill in the art will easily recognize how to make suitable configuration connections between the transceiver and a respective FIFO. Therefore, it is not considered necessary to give a detailed description of each and every signal connection herein.

[0032] Two signal connections, however, should be described in order to gain a more complete understanding of the construction and operation of the transceiver port circuitry 68. An almost full indication signal AF is conventionally asserted by the receive FIFO 66 when the number of empty memory locations is less than, or equal to, a pre-programmed value. Such a condition occurs if data is being written to the FIFO at a rate faster than the FIFO is being read at the media side. Likewise, a read enable signal REN is conventionally asserted to the transmit FIFO 64, and indicates that a receiving device is ready to receive data. In the embodiment of FIGS. 5a and 5b, the AF and REN signals are coupled to the transceiver, with the transceiver receiving AF from the receive FIFO and asserting REN to the transmit FIFO. The transceiver utilizes the AF and REN signals in configuring two overhead bits appended to data words transmitted to the switch. The two overhead bits are used to indicate, among other items, whether the port card receive FIFO buffer has sufficient available space to receive additional data. The overhead bits, therefore, are routed from a receiving transceiver to a transmitting receiver through a reverse crosspoint switch implemented on the switch, in a manner more fully later described.

[0033] Referring now to FIG. 5b, the switch port 54 is connected to a switch matrix or fabric 53. The switch fabric is adapted to route data transmissions, received from a particular transceiver port, to a designated switch port and thence to its corresponding intended recipient transceiver port. This is done under the operational control of an arbitration logic and switch control circuit 55. As referred to previously in the example of FIG. 4, the switch matrix or fabric 53 is conventional in implementation and design, and need not be further described herein. It should be noted, however, that unlike conventional switches implementing a switch fabric, the arbitration logic and switch control circuitry 55 is not implemented as a conventional central control processor. However, a control processor may be used with various

aspects of the present invention.

[0034] Further, flow control decisions are not delayed by using data provided by the transceivers and routed to a conventional central control processor using a dedicated data bus. Instead, overhead bits are sent through essentially a reverse crosspoint switch and appended to the 32-bit data packet (thereby defining a 34-bit transmission character) and used to directly control the flow of information from a transmitter to a receiver, as well as provide other information.

[0035] Turning now to the transceiver port circuitry 68 of FIG. 5a, the transceiver can be viewed as comprising two parts, a transmitter section and a receiver section. When the transceiver is in transmit mode, a 32-bit data word TXIN[31:0] and a 2-bit transmission type word TXTYP[1:0], are clocked out of the transmit FIFO 64 and into a synchronizing parallel input buffer register 70 in accord with 62.5 MHz clock rate.

[0036] Parallel data is clocked out of the buffer register 70 and into a transmit control logic circuit 71. Transmit control logic circuit 71 is responsible for asserting transmission state signals to the transmit FIFO. Such state signals include indications that data packets have been successfully transmitted to all outputs, a beginning-of-packet indication, a retransmission required indication, and the like. Also, transmit control logic 71 is responsible for adaptively reconfiguring TXTYP[1:0] information into a 2-bit flow control overhead bit field when the transceiver is configured to operate in a particular communication mode, designated "overhead-mode" herein. An overhead-mode signal, OH-MODE, is a user programmable state signal, externally sourced, and coupled to the transmit control logic circuitry 71 over an internal communication bus 75.

[0037] In any communication mode, the transmit control logic 71 combines the 32-bit data TXIN[31:0] with either the 2-bit TXTTYP[1:0] or a 2-bit flow control overhead bit field, into a 34-bit data string, the 34-bit data string comprises a transmission word or transmission character. The 34-bit wide transmission characters are serialized by a parallel-to-serial converter 72 (also referred to as a serializer or MUX) and provided to a 2.125 GHz serial output buffer 74. Serialized data is clocked out of the transceiver 68 as a differential signal TXS+/TXS- over a high speed serial transmission line to the input of a corresponding switch port 54. The serial output buffer 74 is clocked by a 2.125 GHz bit clock signal which is, in turn, directly developed by the CRU from an incoming serial data stream sent by the switch port 54. The bit clock signal, BCLK, is a 2.125 GHz strobe which defines the bit cell boundaries of the desired serial data stream. The BCLK signal is directed through a timing generator 86 which comprises divide-by-34 circuitry, such that the 2.125 GHz BCLK signal is divided down to the 62.5 MHz transceiver transmit word clock signal, which is also the transceiver master word clock WCLK signal, in synchronous fashion. Thus, it will be understood that the word boundaries of WCLK and, thus, each 34-bit wide transmission character, will correspond to and be synchronous with every 34th strobe transition edge of the BCLK signal.

[0038] On the receiver side, a high speed serial transmission line is coupled between a high speed output of the switch port 54 and the receiver input of the transceiver circuit 68. The transmission line is configured to provide a differential, serial data stream RXS+/RXS- to the transceiver 68 at a 2.125 Gb/s data rate. The receiver input is coupled to a deserializer, or serial-to-parallel converter 78 which suitably converts the 2.125 GHz serial data stream into 62.5 MHz 34-bit wide parallel transmission characters.

[0039] Serial data is transmitted by the switch port 54 for retrieval by the transceiver circuit 68 without any additional timing reference signals added thereto. A serial stream of data flows over the transmission line with no accompanying clock information. However, the deserializer 78 must process the serial data stream synchronously, such that the resulting 34-bit wide parallel transmission characters are correctly aligned on the appropriate word boundaries. Thus, timing information, i.e., a clock signal, is recovered directly from the serial data stream by a clock recovery unit (CRU) 80. The CRU 80 is a phase and frequency sensitive clock recovery circuit, such as a high-speed phase locked loop (PLL). PLL circuitry suitable for extracting a 2.125 GHz BCLK signal from a 2.125 Gb/s data stream are common circuits implemented in high speed transceiver applications and are well understood by those having skill in the art. Accordingly, it is considered unnecessary to go into detail regarding their construction and operation herein. It is sufficient that CRU 80 is able to recover a bit clock signal BCLK from a serial data stream provided by the switch port 54, and that the recovered BCLK signal is frequency-locked to the frequency of the serial data stream transmitted by the switch port.

[0040] The recovered clock signal, BCLK, is directed through a divide-by-34 timing generator 82 which provides a 62.5 MHz transceiver receive word clock signal to the deserializer 78 and input side of retiming bank 79. The output side of the retiming bank receives the transceiver master word clock signal WCLK, as do receive control logic circuitry 83, a parallel output buffer register 84 and a receiver FIFO 66, from whence received transmission characters are directed to follow-on customer application circuitry over a parallel data interconnect bus. It should be noted that the 62.5 MHz WCLK signal provided to the receive control logic 83, the parallel output buffer 84 and the receiver FIFO 66 is in phase and frequency alignment with the WCLK signal directed to the serializer 72, the parallel input buffer 70 and the transmit FIFO 64, since all of these timing signals derive from the same source, i.e., the timing information recovered from a serial data stream by the CRU 80. In addition, both the transceiver receive and transmit word clock signals are the same frequency as word clock the bit clock signals (BCLK), developed by the CRU 80, and

the incoming serial data stream transmitted by the switch port 54. Because of this relationship between all of the timing signals developed in both the switch port 54 and a corresponding transceiver 68, as well as the bi-directional data transmissions, it will be understood that the switch circuit 50, its composite switch ports 54, and all of their associated transceiver circuits, are frequency-locked to a single clock source, which is developed by a local CMU 89 on the switch card 50, in operational response to a single external timing reference signal.

[0041] Turning now to the switch port 54 illustrated in FIG. 5b, differential, high speed serial data is received from the transceiver's serial output buffer 74 (shown in FIG. 5a) and provided to a data recovery unit (DRU) 86. Because the serial data stream has been clocked out of the transceiver by a BCLK signal which is frequency-locked to the global system clock, the DRU 86 need only be implemented to evaluate the phase of the incoming serial data stream. Frequency lock is maintained by running the DRU 86 off of a BCLK signal directed to the DRU by the CMU 58 which is suitably constructed to multiply a global input, 62.5 MHz WCLK signal by a factor of 34, in order to define a 2.125 GHz BCLK signal suitable for bit clocking operations. The local CMU 58 assists the DRU 86 in obtaining and maintaining frequency lock, such that the DRU need only phase lock to the incoming serial data stream.

[0042] Phase-locked serial data is provided by the DRU 86 to the deserializer 60 which converts the serial data stream into a 34-bit wide parallel transmission character suitable for processing by port logic circuitry 56. The 34-bit wide parallel transmission character is processed by the port logic circuitry 56 in order to determine the intended recipient of the data packet and appropriate switch configuration and connection requests are forwarded to the arbitration logic and switch control circuitry 55, requesting that the switch fabric 53 be configured appropriately. Data traversing the switch fabric 53 towards an intended recipient, is directed into the port logic circuit 56 of the appropriate recipient switch port. Port logic circuitry transfers the 34-bit wide parallel transmission character to the serializer 62 which converts the 62.5 MHz parallel signal into a 2.125 Gb/s serial data stream which is, in turn, clocked out of the switch port through a serial output buffer 94.

[0043] It will be seen from FIG. 5b, that timing information for both the serializer 62 and deserializer 60 is developed through respective timing generators 96 and 98, each responsive to the 2.125 GHz BCLK signal developed by the CMU 58. In addition, the serial output buffer 94 clocks the serial data onto the transmission line and, thence, to the receiver input of the transceiver port circuitry 68 (shown in FIG. 5a), in accordance with a BCLK timing strobe. Thus, it will be seen that the output timing of the serial data stream is defined by a bit clock signal (BCLK) developed by the CMU 58 from a master system clock signal (WCLK). This same bit clock signal (BCLK) is used to frequency-lock the switch port's data recovery unit 86 to an incoming data stream. Therefore the switch port's inputs and outputs are strobed at identical frequencies and the timing boundaries of data received and transmitted are separated only by a transmission line length induced phase shift. Similarly, the BCLK signal which defines the serial data stream frequency is recovered by the clock recovery unit 80 of the transceiver port circuitry 68. The rearward signal is used to define an analog BCLK signal, identical in frequency to the switch port's BCLK frequency, and which is used to strobe the transceiver's serial output buffer 74, thereby defining the frequency of the serial data stream directed to the switch port 54.

[0044] In other words, the embodiment of FIGS. 5a and 5b can be viewed in terms of a transmit/receive clock feedback loop, with the switch port 54 defining a bit cell or bit period clock (BCLK) and using BCLK to define the frequency of a serial data stream directed to a transceiver. The transceiver, in turn, recovers BCLK from the incoming serial data stream and uses this recovered clock to serialize and transmit its serial data streams directed to the switch port 54. Therefore, the serial data stream directed to the input of the switch port is necessarily at the same frequency as the switch port's bit clock (BCLK) and need only be phase-adjusted to accommodate any transmission line length induced phase shift. The present system only requires the serial data streams are encoded to ensure an approximate 15% edge transition density. Moreover, the overhead bits further provide, during much of normal transmission periods, additional edge transitions and thereby increase edge transition density. So long as the serial data stream has the appropriate edge transition density, the system comprising the master system clock (WCLK), in combination with the switch port's CMU 58 and DRU 86, provide a means for ensuring frequency lock and bit alignment between and among a multiplicity of switch ports 54 coupled to a corresponding multiplicity of transceivers 52. This is without regard to variable delays developed through the switch fabric 53 as the fabric changes its switch configuration in response to routing requests, and without the need for a transceiver 68 to burn valuable bandwidth realigning itself to the phase of a new incoming serial data stream.

[0045] Notwithstanding the inherent bit alignment characteristics of the system according to the present invention, it is nevertheless necessary to also provide for some means to word synchronize the information communicated between transceiver 52 and the switch port 54. Even though the transceiver 52 and switch port 54 are frequency-locked together, the transmit and receive data streams may be out of word alignment, with resultant loss of transmission character content. Accordingly, word (or frame) alignment must be established and maintained throughout serial data transmission. To recover word timing, the switch circuit 50 issues particular, pre-defined alignment words to each of the

transceivers 68 during transmission link initialization and handshake protocol establishment.

[0046] Referring now to FIGS. 5a and 5b, and the flow charts of a word alignment and synchronization process illustrated in FIG. 6a, word timing synchronization is established between a switch port and its associated transceiver port by an adaptive feed-back process. In the adaptive feed-back process a predefined alignment word transmitted by the switch are used by the transceivers to establish a transmitter receive word clock. The transceiver then transmits alignment words to the switch, with the switch then comparing the alignment word to an expected alignment word. The switch continues to issue alignment words in the event that the alignment word does not match with the transceiver shifting its transmitted alignment word in bit-by-bit fashion until the transceiver is word synchronized to the switch.

[0047] A flow chart of the word alignment process is illustrated in FIG. 6A. The word alignment process occurs upon power up, reset, or link initialization. The word alignment process executes independently for each transceiver. The word alignment process, therefore, may execute in parallel for any number of transceivers. In step 100 of the word alignment process,the transceiver transmits at least one reset word to the switch. The reset word, comprising all logic "ones" in the described example, requests that the switch begin the initialization and word synchronization process. The receipt of a reset word by the switch causes the switch to transmit alignment words to the transceiver. Alignment words are generated in Step 120 by an alignment word generator and comparator 100 comprising a portion of the port logic circuitry 56. In the example presently described, upon receipt of the reset word, the port logic circuitry 90 causes the alignment word generator and comparator circuit 100 to sequentially generally output alignment words through the serializer 92 and serial output buffer 94. The alignment words are transmitted to the receiver input of the corresponding transceiver port circuitry 68. Alignment words include no inherent data content and so may be devised to contain any form of binary information. Preferably, alignment words are encoded such that the word (frame) boundaries can be easily determined by the alignment word generator and comparator circuit 100. Such an encoding scheme may be implemented in an alignment word comprising a "1" in the LSB and MSB positions of the word, with the remaining bit cells comprising a "0" string, i.e., 10000001, using an 8-bit word as an example. Other bit patterns with increased edge density, such as 10101011, may also be used.

[0048] In step 101 the process determines if the transceiver detects the alignment word. The transceiver circuitry which accomplishes this deserializes alignment words using the deserializer 78 and uses an alignment detector circuit 102, coupled to "snoop" the parallel bus coupled between the deserializer 78 and the transceiver's retiming register bank.

[0049] If the alignment detector does not detect the correct alignment word the alignment detector provides a signal to the receive word clock timing generator to shift the receive word clock by one bit in step 102. Examination of the received alignment words and, if necessary, the shifting of the receive word clock continues until the alignment detector detects the correct alignment word. Once the receive word clock is correctly aligned, the transceiver begins transmitting alignment words to the switch in step 103. The switch, using the alignment word generator and comparator circuit 100 compares the received alignment words to the expected alignment word in step 104. If the switch detects the correct alignment word the switch in step 107 issues IDLE word to the transceiver to signal that the transmitter is now word synchronized with the switch. The process then returns.

[0050] If the transceiver continues to receive alignment words, the alignment detection circuit 102 causes the frame generator and bit shifter to shift its transmit word boundary by one bit position after receipt of every 32 alignment words from the switch in steps 105 and 106. The process repeats until the alignment word generator and comparator circuit 100 in the switch determines that the alignment words sent by the transceiver are correctly framed in accordance with the switch word clock signal.

[0051] Bit and word alignment is accomplished only during link initialization. Since this process occurs relatively infrequently, the system of the present invention is not required to support fast phase acquisition and need only maintain frequency lock in the manner described above. Moreover, any small variation in phase of the signal received by the switch, whether due to component aging or temperature variations, is accounted for by the data recovery unit (DRU) of the switch circuitry. In addition, since a master reference clock is provided on the switch circuit, the system does not require transmission characters to incorporate additional overhead bits devised to absorb the types of bit loss that can occur with multiple reference clocks driving multiple transceivers, as is common in prior art implementations.

[0052] Although ensuring that the switch and all of its attendant transceiver circuits are bit and word synchronized, the synchronization method does require some means to ensure that both transmit and receive serial data streams contain sufficient signal edge transition density in order to maintain the established frequency lock. Given a preferred 15% edge transition density in a serial data stream, it will be understood that there need only be five transition edges incorporated in a 34-bit transmission character. Accordingly, only a few overhead bits are needed for each transmission character (word or frame) in order to maintain synchronization and frequency lock. Using this approach, the effective data bandwidth is reduced by only approximately 6%, as compared to a 20 to 40% bandwidth reduction in prior art systems. The higher bandwidth reduction is needed

in prior art systems in order to guarantee that the data signal contains a sufficiently high transition rate for fast phase recovery.

[0053] As opposed to merely taking up transmission bandwidth, the overhead bits referred to above enable a particularly advantageous feature to be realized by the system. In contrast to the conventional prior art approach, which required all transceiver port data transmissions and connection requests to flow through a data bus to a single central control point, such as a control processor, the system implements flow control by changing the conventional meaning and function of the two overhead bits appended to the 32-bit data packet. The overhead bits appended to the 32-bit data packet are able to provide both low level flow control information and acknowledge and other information. The self-routing character switching and low level flow control architecture, as well as self-routing data switching, will now be described with reference to FIGS. 5a, 5b and FIG. 7.

[0054] With particular reference to FIG. 7, there is shown a number of 34-bit transmission characters, as those characters would appear at the TXS transmit output and the RXS receiver input of the transceiver circuit 68. A 34-bit transmission character including a 32-bit data word, a command word, a connection request (CRQ), such as would be sent by the transceiver to its corresponding switch port, and a connection request reply, such as would be sent by a switch port back to its corresponding transceiver are illustrated in FIG. 7.

[0055] The data word format is the format of a data word as it would be seen on the serial data lines spanning a transceiver and its corresponding switch port. The data word format is a 34-bit transmission character comprising a 32-bit data payload 110 and a 2-bit overhead field 112 appended thereto in the MSB positions. As was mentioned above, the 2-bit overhead field 112 has a conventional usage, whereby if the overhead mode (OH-MODE) signal, coupled to the transmit control circuit 71 and the receive control circuit 83, is in a first state, the overhead bits contained in the overhead field 112 identify the transmission signal's transmission type (TXTYP[1:0]), as well as identifying the receive signal's receive type (RXTYP[1:0]). In contrast, when the overhead mode (OH-MODE) signal is in a second state, the bit content of the overhead bit field 112 contains flow control channel information which is time-shared with the signaling between the switch and a transceiver for acknowledgment and response bits. The main application for this flow control channel is to prevent the receive FIFO of the receiving transceiver port from overflowing and, by using this flow control channel when the receive FIFO is almost full, transmitting transceiver port will be disabled from sending further data.

[0056] To provide self-routing and message passing functions, the transceiver and switch require different data types to differentiate between data words, connection request words, message words or command words. Depending on the state of the OH-MODE signal, and thus the mode that the transceiver is in, different data types are recognized by the transceiver. These data types at the transceiver/FIFO parallel interface are encoded in the TXTYP[1:0] or RXTYP[1:0] bits. The transmit control circuit 71 and receive control circuit 83 respectively encode these data types into the two over head bits B[1:0] or decode the two overhead bits B[1:0] for passage to the receive FIFO.

[0057] The command word format, as seen on the serial data lines at the output of the transceiver or the switch, comprises a 2-bit overhead field 114 appended to a 32-bit string 116 at the MSB positions. The overhead field 114 comprises two overhead bits (0,0) which are added by the transceiver or switch to designate a command word to the receiving switch or transceiver. The next three bit positions (from MSB to LSB) begin with a hard coded "1", followed by a pair of bits B[1:0] that are encoded to define an alignment word, one of two flow control channels or an acknowledge signal or link initialization reset signal. An additional 5-bit field C[4:0] is reserved for defining a command type and is used to identify alignment words, an IDLE signal, a RESET, signal, and the like. The command type field is followed by an optional 16-bit data payload field D[15:0], followed by an alternating pattern of "ones" and "zeros", terminating in a parity bit in the LSB position.

[0058] A connection request (CRQ) command word format as seen at the transceiver-to-switch interface includes an overhead bit field 118 appended to a 32-bit CRQ command word 120. The overhead bit field 118 comprises two overhead bits that are added by the transceiver's transmit control circuit 71 in order to designate either a CRQ word to the receiving switch, when the bits are configured (0,0) or to designate a header word for the next packet when the bits are configured (1,1). Similarly, a connection request as seen at the switch-to-transceiver interface includes an overhead field 122 appended to a CRQ command word 124. Two overhead bits are added by the switch in order to designate a command word to the transceiver when the bits are configured (0,0). This word is provided by the switch to the receiving port, when an ACK signal is asserted to the transmitting port, and functions as start-of-packet. The switch-to-transceiver CRQ word format contains the current active connections for the transmitting port in a 16-bit active connection field C[15:0].

[0059] Thus, in accordance with the invention, a command word can be used to send a connection request (CRQ) from a particular transceiver port to the switch matrix. An acknowledgment (ACK) to the request is returned from the switch to the requesting transceiver port by appending two overhead bits, configured (1,1) in either a command word's overhead field 114 or a data word's overhead field 112. Flow control channel information is also shared between a receiving transceiver port and a transmitting transceiver port by reconfiguring the two overhead bits comprising the overhead bit fields

112 and 114 of a data or command word. As will be described in greater detail below, a ready-to-received (RTR) configuration signal informs the transmitting transceiver port that there is sufficient room in the receiving transceiver port's receive FIFO and that it is appropriate to continue to transmit data. A not-ready-to-receive (NRTR) signal informs the transmitting transceiver port that the receiving transceiver port's receive FIFO is filling up and it is, therefore, not appropriate to continue sending data.

[0060] A ready-to-receive (RTR) signal is generated by configuring the overhead bits as (0,1) while a not-ready-to-receive (NRTR) signal is generating by configuring the overhead bits as (1,0). When the overhead bits are configured as (1,1), as indicated above, the overhead bits comprise an acknowledge (ACK) signal which is returned from the switch to a transceiver port which has made a connection request. It is the function of the switch, particularly the port logic 90 and arbitration logic and switch control circuit 55 to either generate the appropriate overhead bits (such as ACK) or to intercept flow control messages (such as RTR and NRTR) and re-direct them to the appropriate transceiver port so that effective flow control is maintained.

[0061] In order to better understand the utility of the overhead bits, it will be useful to consider how flow control overhead bits are generated by a transceiver in response to a FIFO almost full condition, and how flow control overhead bits are used in order to prevent further transmission. Referring now to FIGS. 5a and 5b, the receive FIFO 66 is conventionally provided with a signal line that indicates that the FIFO is in an almost full condition. The receive FIFO 66 can overflow if data arrives and is written to the FIFO faster than it can be read from the FIFO to the user's application circuit or physical media. In the example of FIG. 5a, the almost full signal AF from the receive FIFO 66 is coupled to the transceiver's input register 70 which passes the AF signal to the transceiver's transmit control circuit 71. The AF signal will be asserted when the number of empty FIFO locations is less than or equal to a value pre-programmed into the FIFO. This minimum value typically depends on the latency period between the time when the AF signal is asserted and when it is received at the transmitting transceiver. The latency period is such that 14 + N more words are able to be transmitted, where N relates to a distance latency parameter and depends on the physical distance between the transceivers and the switch. It will be understood by those having skill in the art how to calculate the number of words that might be written into an almost full FIFO during a latency period and how to program this value into the FIFO such that AF is asserted at the proper time.

[0062] Transmit control circuit 71 receives the almost full indication and, if put into overhead mode by the appropriate OH-MODE signal, appends a not-ready-to-receive (NRTR) signal to a command or data word. As mentioned above, an NRTR signal is generated by configuring the overhead bits as (1,0). The command or data word including the NRTR signal is transmitted to the switch which recognizes the overhead bits as not comprising a conventional pattern and, in response, strips the NRTR signal from the command or data word and routes it to the appropriate transceiver for action. The overhead bits are recognized by the switch port's port logic circuit 56 which directs them to the switch's arbitration logic and switch control circuit 55 through its parallel bus connection through a plurality of 16 to 1 MUXes 91 (only one of which is shown for clarity). The 16 to 1 MUXes provide essentially the functions of a reverse cross-point switch, which is how the described function is implemented in one embodiment. Arbitration logic and switch control circuit 55 recognizes which of the 16 switch ports provided the NRTR signal and, since it is in control of the configuration of the fabric 53, the arbitration logic and switch control circuit 55 understands which of the 16 switch ports is coupled to the transceiver which is transmitting data to the almost full recipient. Arbitration logic and switch control circuit 55 then provides the NRTR signal (1,0) to the appropriate switch port's port logic circuit 56 through MUX 91. That switch port's port logic circuit 56 appends the NRTR overhead bits to the next outgoing transmission to its corresponding transceiver 68.

[0063] In the transceiver port circuitry 68, the (1,0) overhead bits are directed to the receive control circuitry 83, as described above, which recognizes that the (1,0) overhead bit pattern represents a not-to-ready to receive condition and that the transceiver should cease transmitting. Receive control circuit 83 communicates with transmit control logic circuit 71 over the internal communication bus 75 and passes the overhead signal values to the transmit control logic circuit for command processing. In transmit control circuit, the received flow control signal is ANDed with a read enable signal, controlled by the transmit control logic. The resulting read enable signal REN is connected through the input register 70 to the transmit FIFO 64. When a not-to-ready to receive signal is ANDed with read enable, the resulting REN signal is de-asserted, instructing the transmit FIFO 64 that data reading is no longer enabled. Data transmission thereby ceases. Thus, the transceiver is only able to read a word from the transmit FIFO 64 if the flow control signal (the overhead bits) from the receiving transceiver are in a ready-to-receive (RTR) state. It should be noted, herein, that the switch is also able to independently assert and append an NRTR overhead pattern to any particular transmitting transceiver, in order to force an IDLE into the data stream to the transceiver whenever needed.

[0064] Turning now to FIG. 8, there is shown a semi-schematic block diagram of an embodiment of the flow control feedback mechanism in accordance with the present invention.

[0065] In the embodiment of FIG. 8, information is being simultaneously transmitted and received by three

transceiver port cards 130, 132 and 134, respectively indicated as port cards A, B and C, through a switch matrix unit 50. In the example, serial data streams are being transmitted by transceiver card A to transceiver card B; from transceiver card B to transceiver card C; and, from transceiver card C to transceiver card A. For purposes of the example, it is assumed that the receive FIFO in transceiver card C is filling up. Accordingly, transceiver card C must have some means of signaling its data transmission partner, transceiver card B to temporarily cease transmitting data. Transceiver card C appends the overhead bits comprising the NRTR flow control code (1,0) to the 32-bit data packet comprising its next transmission to the switch matrix. The arbitration logic and switch control circuitry (55 of FIG. 5), in combination with the port logic circuitry (56 of FIG. 5) evaluates the overhead bits from transceiver card C and recognizes that transceiver card C wishes to alert its transmission partner that it is no longer ready to receive data. Since all of the current switch matrix connections are made through the switch control and port logic circuitry, the switch matrix is able to identify the current transmission partner of transceiver card C as transceiver card B and is able to re-direct the NRTR signal to transceiver card B by stripping the overhead bits from the transceiver card C transmission character and re-appending them to the next transmission character directed to transceiver card B, i.e., onto the data stream coming from transceiver card A to transceiver card B. In this manner, transceiver card B is made aware that its transmission partner, i.e., the intended recipient of its transmitted data, is no longer ready to receive transmission characters and transceiver card B must temporarily stop sending data.

[0066] Using the overhead bits to provide flow control for data passing through the switch decreases utilization of the data bus used in conventional prior art-type implementations with a resulting increase in switch bandwidth. In addition, use of the round robin arbitration and switch control logic allows the data bus and processor of prior art implementations to be completely eliminated, along with the congestion associated with centralized control architecture.

[0067] To summarize, the switch port's transmitter sends out data words that come from the switch matrix, adding the appropriate overhead bit information for acknowledges, response bits and flow control. Acknowledges are used to signal a corresponding transceiver that a connection request has been granted. Response bits are used with a Multi-Queue connection request word, which need not be considered further herein. The flow control channel is used to pass state information from the receiving transceiver port to the transmitting transceiver port. The switch redirects the flow control signals to the correct output using the current switch connection state information contained within an arbitration logic and switch control circuit. Thus, the self routing and flow control architecture in accordance with the present invention, enables a significant improvement in overall system bandwidth utilization and significantly eases the task of the system designer in finding sufficient integrated circuit chip real estate to accommodate the circuitry necessary to perform all the requisite tasks.

Claims

1. A high speed network switching apparatus, comprising:

   a switch including:

   a plurality of switch ports (54), each switch port (54) adapted to transmit and receive a high-speed serial data stream; and
   a switch fabric (53) coupled to the plurality of switch ports (54), the switch fabric (53) routing data among and between the switch ports (54);

   a plurality of transceiver circuits (52), each transceiver circuit (52) configured to transmit and receive a high speed serial data stream between a corresponding one of the plurality of switch ports (54) so as to establish a transmission channel between a corresponding transmitting transceiver circuit (68, TX) and a corresponding receiving transceiver circuit (68, RX);

   wherein the data stream includes command and data words comprising:

   a data portion (110, 116, 120, 124); and
   a header portion (112, 114, 118, 122), **characterised in that** the header portion (112, 114, 118, 122) includes overhead bits configured to provide a ready-to-receive indication (RTR) from a receiving transceiver circuit (68, RX) to a transmitting transceiver circuit (68, TX) when the overhead bits are in a first binary sequence, a not-ready-to-receive (NRTR) indication from a receiving transceiver (68, RX) to a transmitting transceiver (68, TX) when the overhead bits are in a second binary sequence, the switch adaptively routing the overhead bits from the corresponding receiving transceiver circuit (68, TX) to the corresponding transmitting receiver circuit (68, RX).

2. The high speed network switching apparatus of claim 1 wherein the switch adaptively routes overhead bits from the corresponding receiving transceiver circuit (68, RX) to the corresponding transmitting transceiver circuit (68, TX) using a reverse switch fabric.

3. The high speed network switching apparatus of claim 2 further comprising means for routing the overhead bits to the reverse switch fabric and for routing the data portion (110, 116, 120, 124) to the switch fabric (53).

4. The high speed network switching apparatus of at least one of the preceding claims, further comprising a transmit data buffer (64, TXFIFO) coupled to the transceiver (68) over a parallel interface, the interface defining at least an enable signal for enabling parallel data to be read to the transceiver (68, TX) when the signal is in a first state, and for disabling parallel data from being read to the transceiver (68) when the signal is in a second state.

5. The high speed network switching apparatus of at least one of the preceding claims, further comprising a receive data buffer (66, RXFIFO) coupled to the transceiver (68) over a parallel interface, the interface defining at least an indication signal when the data buffer is almost full.

6. The high speed network switching apparatus of claim 5 further comprising means for appending an overhead bit field (110, 116, 120, 124) to a data or command word (110, 116, 120, 124), the overhead bit field (110, 116, 120, 124) containing overhead bits having a first configuration when the almost full indication signal is asserted, the overhead bits having a second configuration when the almost full indication signal is not asserted.

7. The high speed network switching apparatus of claim 5 or 6 further comprising means for reading an appended overhead bit field (110, 116, 120, 124), the means for reading asserting the read enable signal to the first, enable, state when the overhead bits are in the second configuration, and asserting the read enable signal to the second, disable, state when the overhead bits are in the first configuration.

8. The high speed network switching apparatus of at least one of the preceding claims wherein the overhead bits transmitted by a particular switch transmit port to a particular transceiver indicate acknowledgment of receipt by the particular switch receive port of a request by the particular transceiver.

9. The high speed network switching apparatus of at least one of the preceding claims wherein the overhead bits transmitted by a particular switch transmit port to a particular transceiver indicate the granting of a connection request requested by the particular transceiver.

Patentansprüche

1. Schnelle Netzwerk-Switch-Vorrichtung, umfassend:

einen Switch, der folgendes enthält:

mehrere Switch-Ports (54), wobei jeder Switch-Port (54) so ausgelegt ist, daß er einen schnellen seriellen Datenstrom sendet und empfängt; und

ein an die mehreren Switch-Ports (54) angekoppeltes Koppelfeld (53), wobei das Koppelfeld (53) Daten unter und zwischen den Switch-Ports (54) routet;

mehrere Sender-/Empfängerschaltungen (52), wobei jede Sender-/Empfängerschaltung (52) so konfiguriert ist, daß sie einen schnellen seriellen Datenstrom zwischen einem entsprechenden der mehreren Switch-Ports (54) sendet und empfängt, um so einen Übertragungskanal zwischen einer entsprechenden sendenden Sender-/Empfängerschaltung (68, TX) und einer entsprechenden empfangenden Sender-/Empfängerschaltung (68, RX) herzustellen;

wobei der Datenstrom Befehls- und Datenwörter enthält, die folgendes umfassen:

einen Datenteil (110, 116, 120, 124); und
einen Kopfteil (112, 114, 118, 122), **dadurch gekennzeichnet, daß** der Kopfteil (112, 114, 118, 122) Overhead-Bit enthält, die so konfiguriert sind, daß sie folgendes bereitstellen: eine Empfangsbereitschaftsanzeige (RTR) von einer empfangenden Sender-/Empfängerschaltung (68, RX) zu einer sendenden Sender-/Empfängerschaltung (68, TX), wenn sich die Overhead-Bit in einer ersten binären Sequenz befinden, eine Nicht-Empfangsbereitschaftsanzeige (NRTR) von einem empfangenden Sender-/Empfänger (68, RX) zu einem sendenden Sender-/Empfänger (68, TX), wenn sich die Overhead-Bit in einer zweiten binären Sequenz befinden, wobei der Switch die Overhead-Bit aus der entsprechenden empfangenden Sender-/Empfängerschaltung (68, TX) adaptiv zu der entsprechenden sendenden Sender-/Empfängerschaltung (68, RX) routet.

2. Schnelle Netzwerk-Switch-Vorrichtung nach Anspruch 1, wobei der Switch Overhead-Bit aus der entsprechenden empfangenden Sender-/Empfängerschaltung (68, RX) adaptiv zu der entsprechen-

den sendenden Sender-/Empfängerschaltung (68, TX) unter Verwendung eines umgekehrten Koppelfeldes routet.

3. Schnelle Netzwerk-Switch-Vorrichtung nach Anspruch 2, weiterhin mit Mitteln zum Routen der Overhead-Bit zu dem umgekehrten Koppelfeld und zum Routen des Datenteils (110, 116, 120, 124) zu dem Koppelfeld (53).

4. Schnelle Netzwerk-Switch-Vorrichtung nach mindestens einem der vorhergehenden Ansprüche, weiterhin mit einem über eine parallele Schnittstelle an den Sender-/Empfänger (68) angekoppelten Sendedatenpuffer (64, TXFIFO), wobei die Schnittstelle mindestens ein Freigabesignal zum Freigeben des Auslesens paralleler Daten an den Sender-/Empfänger (68, TX), wenn sich das Signal in einem ersten Zustand befindet, und zum Sperren des Auslesens paralleler Daten an den Sender-/Empfänger (68), wenn sich das Signal in einem zweiten Zustand befindet, definiert.

5. Schnelle Netzwerk-Switch-Vorrichtung nach mindestens einem der vorhergehenden Ansprüche, weiterhin mit einem über eine parallele Schnittstelle an den Sender-/Empfänger (68) angekoppelten Empfangsdatenpuffer (66, RXFIFO), wobei die Schnittstelle mindenstens ein Anzeigesignal, wann der Datenpuffer fast voll ist, definiert.

6. Schnelle Netzwerk-Switch-Vorrichtung nach Anspruch 5, weiterhin mit Mitteln zum Anhängen eines Overhead-Bit-Feldes (110, 116, 120, 124) an ein Daten- oder Befehlswort (110, 116, 120, 124), wobei das Overhead-Bit-Feld (110, 116, 120, 124) Overhead-Bit enthält, die eine erste Konfiguration aufweisen, wenn das Fast-voll-Anzeigesignal gesetzt ist, und die Overhead-Bit eine zweite Konfiguration aufweisen, wenn das Fast-voll-Anzeigesignal nicht gesetzt ist.

7. Schnelle Netzwerk-Switch-Vorrichtung nach Anspruch 5 oder 6, weiterhin mit Mitteln zum Lesen eines angehängten Overhead-Bit-Feldes (110, 116, 120, 124), wobei die Mittel zum Lesen das Lesefreigabesignal in den ersten Freigabe-Zustand versetzen, wenn sich die Overhead-Bit in der zweiten Konfiguration befinden, und das Lesefreigabesignal in den zweiten Sperr-Zustand versetzen, wenn sich die Overhead-Bit in der ersten Konfiguration befinden.

8. Schnelle Netzwerk-Switch-Vorrichtung nach mindestens einem der vorhergehenden Ansprüche, wobei die durch einen bestimmten Switch-Sendeport zu einem bestimmten Sender/Empfänger gesendeten Overhead-Bit bestätigen, daß der bestimmte Switch-Empfangsport eine Anforderung des bestimmten Sender/Empfängers empfangen hat.

9. Schnelle Netzwerk-Switch-Vorrichtung nach mindestens einem der vorhergehenden Ansprüche, wobei die durch einen bestimmten Switch-Sendeport zu einem bestimmten Sender/Empfänger gesendeten Overhead-Bit die Gewährung einer von dem bestimmten Sender/Empfänger angeforderten Verbindungsanforderung anzeigen.

Revendications

1. Appareil de commutation de réseau à grande vitesse, comprenant :

un commutateur comprenant :

une pluralité de ports de commutation (54), chaque port de commutation (54) étant adapté pour transmettre et recevoir un flux de données série à grande vitesse ; et
une matrice de commutation (53) couplée à la pluralité de ports de commutation (54), la matrice de commutation (53) acheminant des données au sein de et entre les ports de commutation (54) ;
une pluralité de circuits d'émission/réception (52), chaque circuit d'émission/réception (52) étant configuré pour transmettre et pour recevoir un flux de données série à grande vitesse avec l'un des ports de commutation correspondants de la pluralité de ports de commutation (54), de manière à établir un canal de transmission entre un circuit d'émission correspondant (68, TX) et un circuit de réception correspondant (68, RX) ;

dans lequel le flux de données comprend des mots de commande et de données comprenant :

une partie de données (110, 116, 120, 124) ; et
une partie d'en-tête (112, 114, 118, 122),

caractérisé en ce que la partie d'en-tête (112, 114, 118, 122) comprend des bits d'en-tête configurés pour transmettre une indication prête à recevoir (RTR) d'un circuit de réception (68, RX) à un circuit d'émission (68, TX) lorsque les bits d'en-tête sont dans une première séquence binaire, une indication non prête à recevoir (NRTR) d'un circuit de réception (68, RX) à un circuit d'émission (68, TX) lorsque les bits d'en-tête sont dans une seconde séquence binaire, le commutateur acheminant de manière adaptative les bits d'en-tête du circuit

de réception correspondant (68, TX) au circuit de transmission correspondant (68, RX).

2. Appareil de commutation de réseau à grande vitesse selon la revendication 1, dans lequel le commutateur achemine de manière adaptative les bits d'en-tête du circuit de réception correspondant (68, RX) au circuit de transmission correspondant (68, TX) en utilisant une matrice de commutation inversée.

3. Appareil de commutation de réseau à grande vitesse selon la revendication 2, comprenant en outre un moyen pour acheminer les bits d'en-tête jusqu'à la matrice de commutation inversée, et pour acheminer la partie de données (110, 116, 120, 124) jusqu'à la matrice de commutation (53).

4. Appareil de commutation de réseau à grande vitesse selon au moins l'une des revendications précédentes, comprenant en outre une mémoire tampon de transmission de données (64, TXFIFO) couplée à l'émetteur/récepteur (68) sur une interface parallèle, l'interface définissant au moins un signal de validation destiné à permettre aux données parallèles d'être lues par l'émetteur/récepteur (68, TX) lorsque le signal est dans un premier état, et destiné à désactiver les données parallèles afin qu'elles ne soient plus lues par l'émetteur/récepteur (68) lorsque le signal est dans un second état.

5. Appareil de commutation de réseau à grande vitesse selon au moins l'une des revendications précédentes, comprenant en outre une mémoire tampon de réception de données (66, RXFIFO) couplée à l'émetteur/récepteur (68) sur une interface parallèle, l'interface définissant au moins un signal d'indication lorsque la mémoire tampon est presque pleine.

6. Appareil de commutation de réseau à grande vitesse selon la revendication 5, comprenant en outre un moyen pour joindre un champ de bits d'en-tête (110, 116, 120, 124) à un mot de données ou de commande (110, 116, 120, 124), le champ de bits d'en-tête (110, 116, 120, 124) contenant des bits d'en-tête possédant une première configuration lorsque le signal d'indication de mémoire presque pleine est confirmé, et les bits d'en-tête possédant une seconde configuration lorsque le signal d'indication de mémoire presque pleine n'est pas confirmé.

7. Appareil de commutation de réseau à grande vitesse selon la revendication 5 ou 6, comprenant en outre un moyen pour lire un champ de bits d'en-tête joint (110, 116, 120, 124), le moyen de lecture confirmant le signal de validation de lecture dans le premier état de validation lorsque les bits d'en-tête sont dans la seconde configuration, et confirmant le signal de validation de lecture dans le second état d'invalidation lorsque les bits d'en-tête sont dans la première configuration.

8. Appareil de commutation de réseau à grande vitesse selon au moins l'une des revendications précédentes, dans lequel les bits d'en-tête transmis par un port d'émission de commutation particulier vers un émetteur/récepteur particulier indiquent la confirmation de réception par le port de réception de commutation particulier d'une requête effectuée par l'émetteur/récepteur particulier.

9. Appareil de commutation de réseau à grande vitesse selon au moins l'une des revendications précédentes, dans lequel les bits d'en-tête transmis par un port d'émission de commutation particulier vers un émetteur/récepteur particulier indiquent l'octroi d'une demande de connexion effectuée par l'émetteur/récepteur particulier.

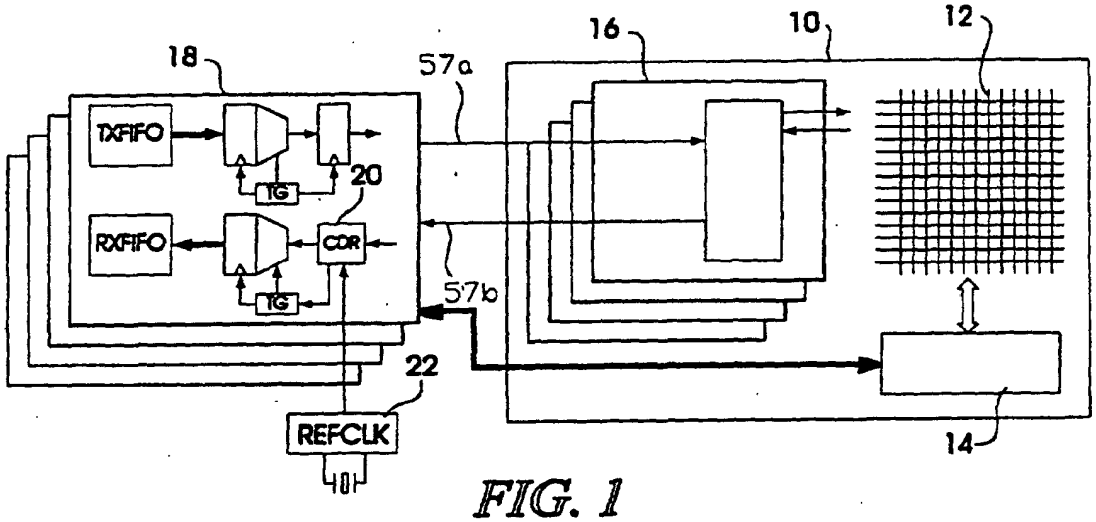

18
16
10
12
57a
TXFIFO
TG
20
RXFIFO
CDR
TG
57b
22
REFCLK
40
14

FIG. 1

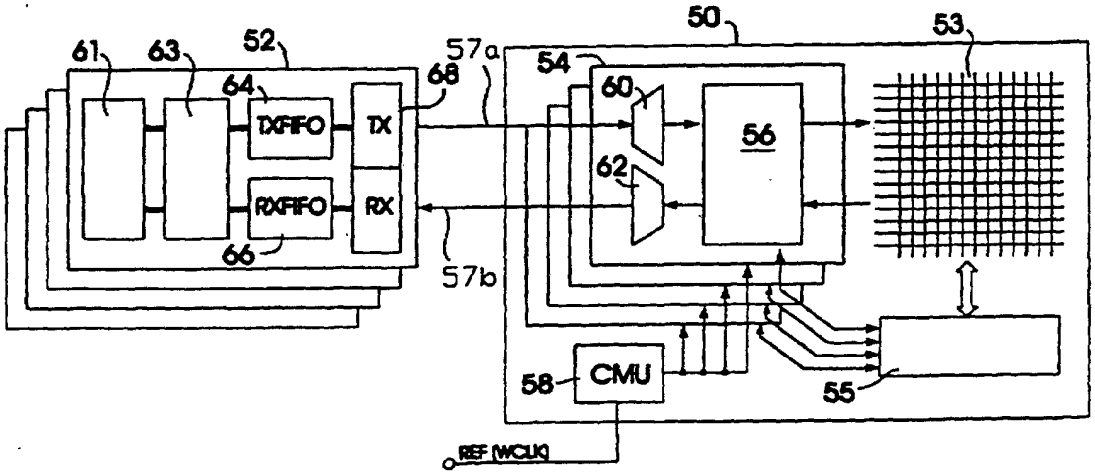

61
63
52
50
53
57a
54
64
68
60
TXFIFO
TX
56
62
RXFIFO
RX
66
57b
58
CMU
55
REF (WCLK)

FIG. 4

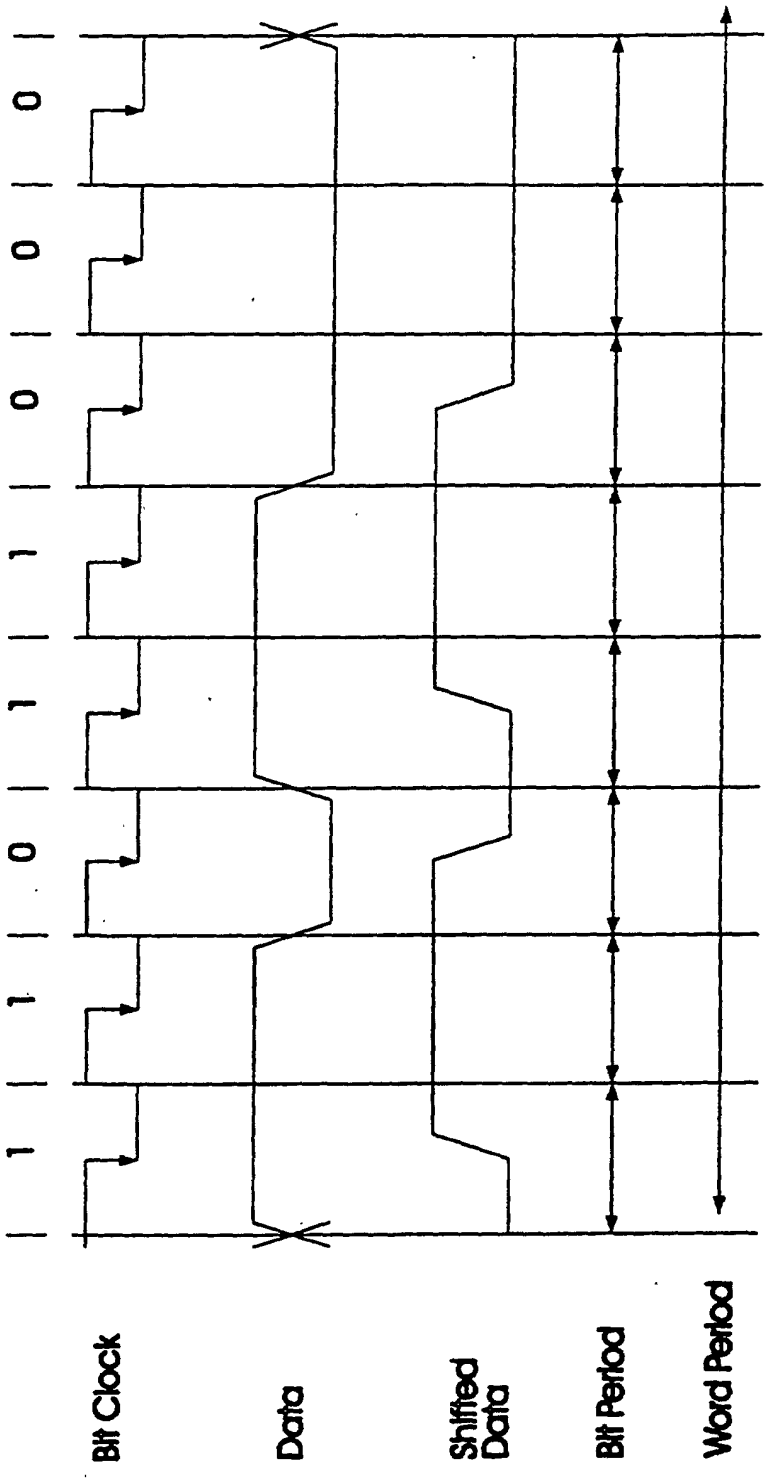

1
1
0
1
1
0
0
0
Bit Clock
Data
Shifted Data
Bit Period
Word Period

FIG. 2

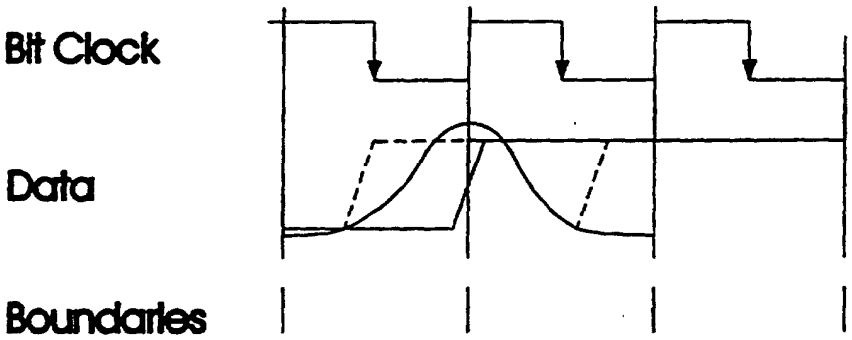


FIG. 3

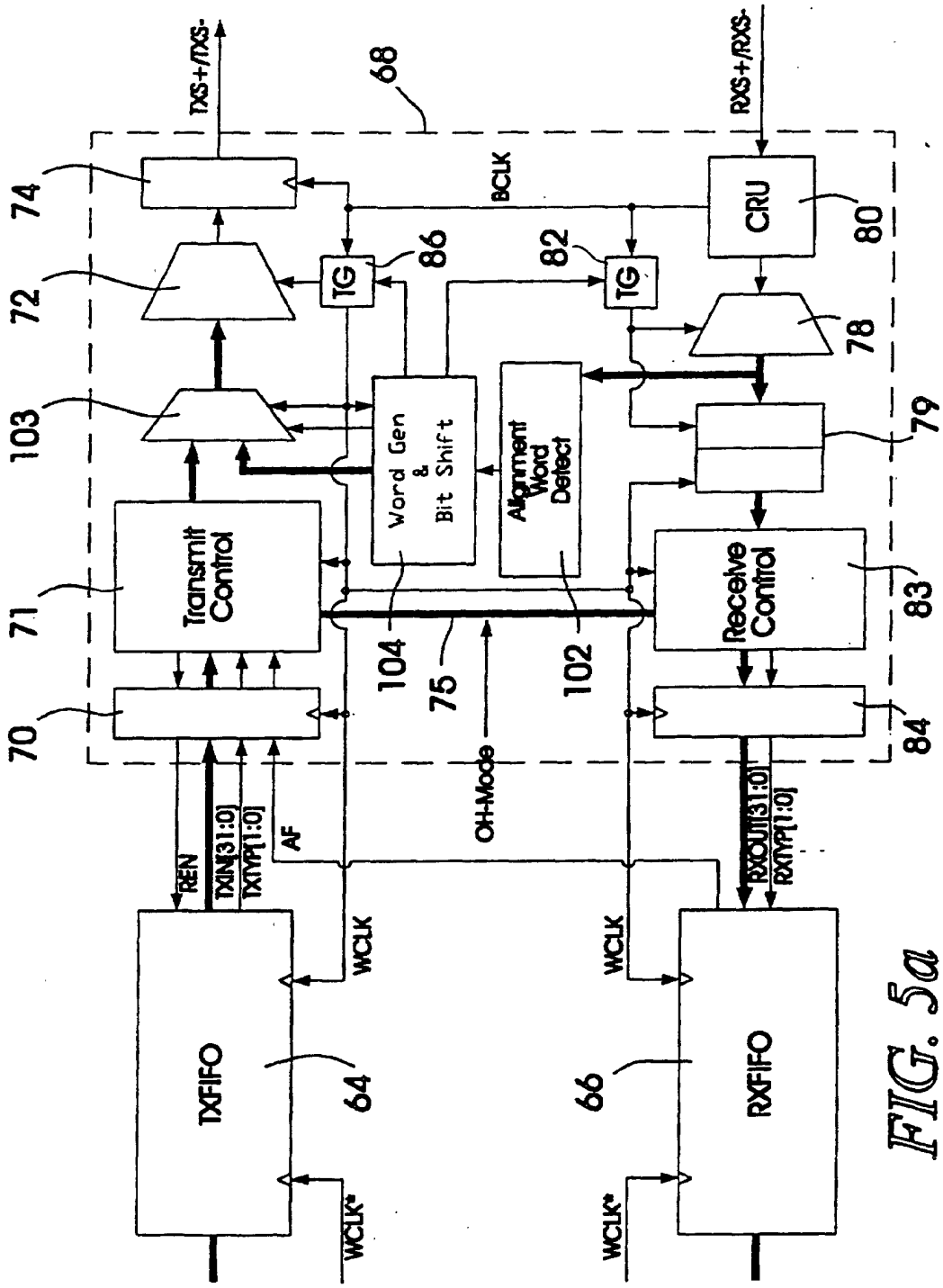

TXS+/TXS-
RXS+/RXS-
68
74
72
103
71
70
BCLK
TG
86
82
TG
CRU
80
78
79
Word Gen & Bit Shift
Alignment Word Detect
Transmit Control
Receive Control
83
84
104
75
102
OH-Mode
REN
TXIN[31:0]
TXTYP[1:0]
AF
RXOUT[31:0]
RXTYP[1:0]
WCLK
WCLK
TXFIFO
64
66
RXFIFO
WCLK*
WCLK*

FIG. 5a

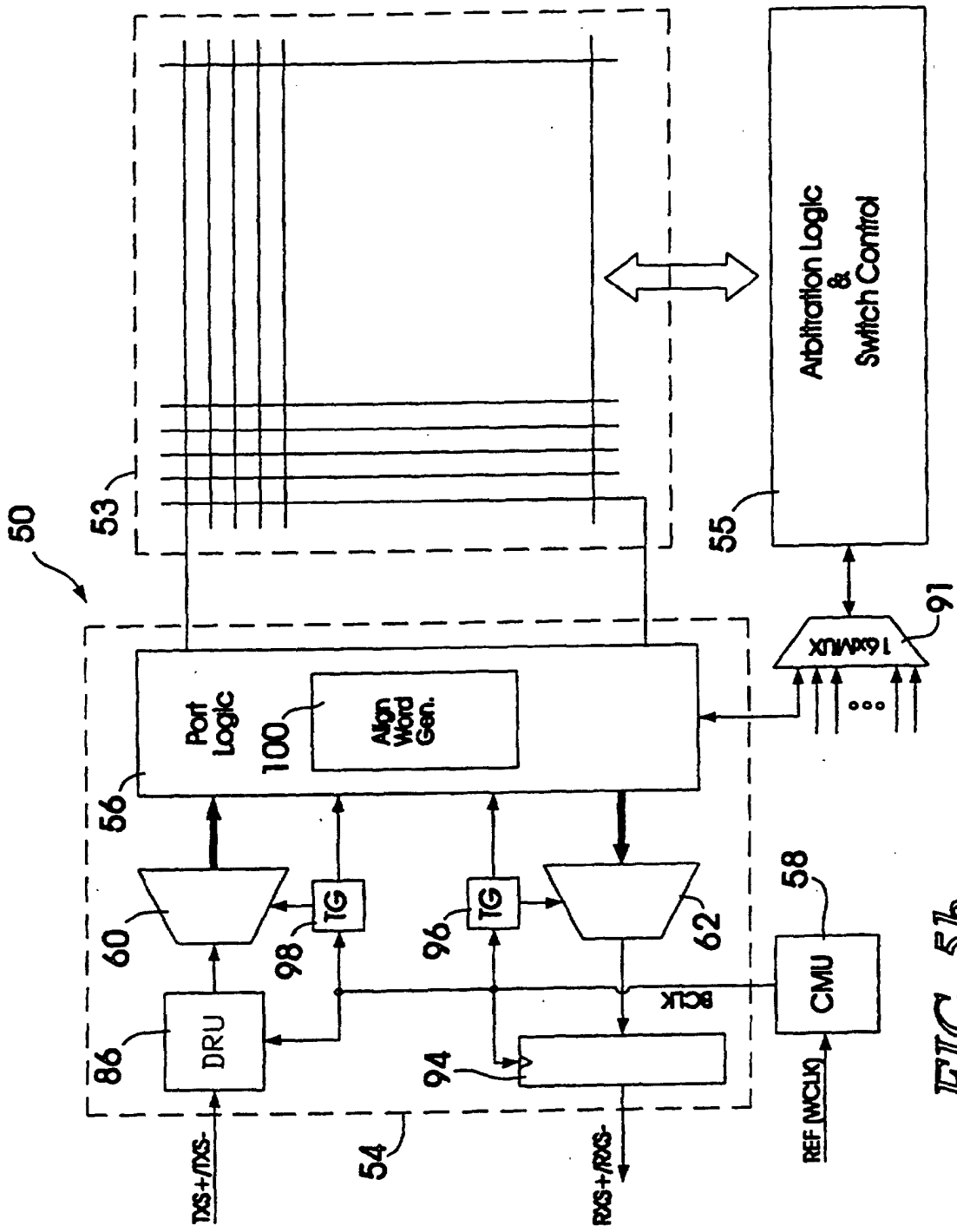
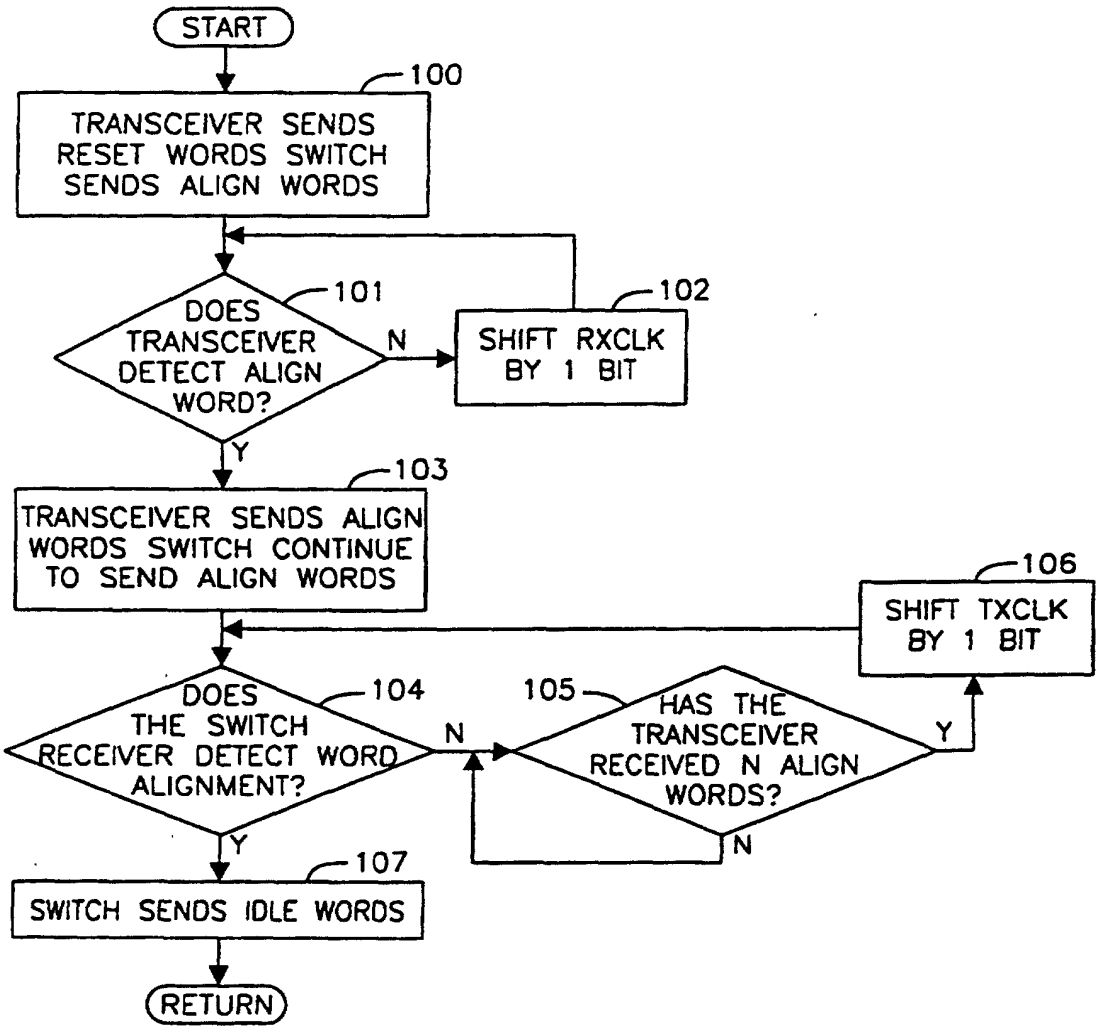

50
53
55
Arbitration Logic & Switch Control
91
16xMUX
Port Logic
100
Align Word Gen.
56
60
TG
98
96
TG
62
58
CMU
RCLK
DRU
86
94
54
TXS+/TXS-
RXS+/RXS-
REF (WCLK)

FIG. 5b

FIG.6


START
100
TRANSCEIVER SENDS RESET WORDS SWITCH SENDS ALIGN WORDS
101
DOES TRANSCEIVER DETECT ALIGN WORD?
N
102
SHIFT RXCLK BY 1 BIT
Y
103
TRANSCEIVER SENDS ALIGN WORDS SWITCH CONTINUE TO SEND ALIGN WORDS
106
SHIFT TXCLK BY 1 BIT
104
DOES THE SWITCH RECEIVER DETECT WORD ALIGNMENT?
N
105
HAS THE TRANSCEIVER RECEIVED N ALIGN WORDS?
Y
N
Y
107
SWITCH SENDS IDLE WORDS
RETURN

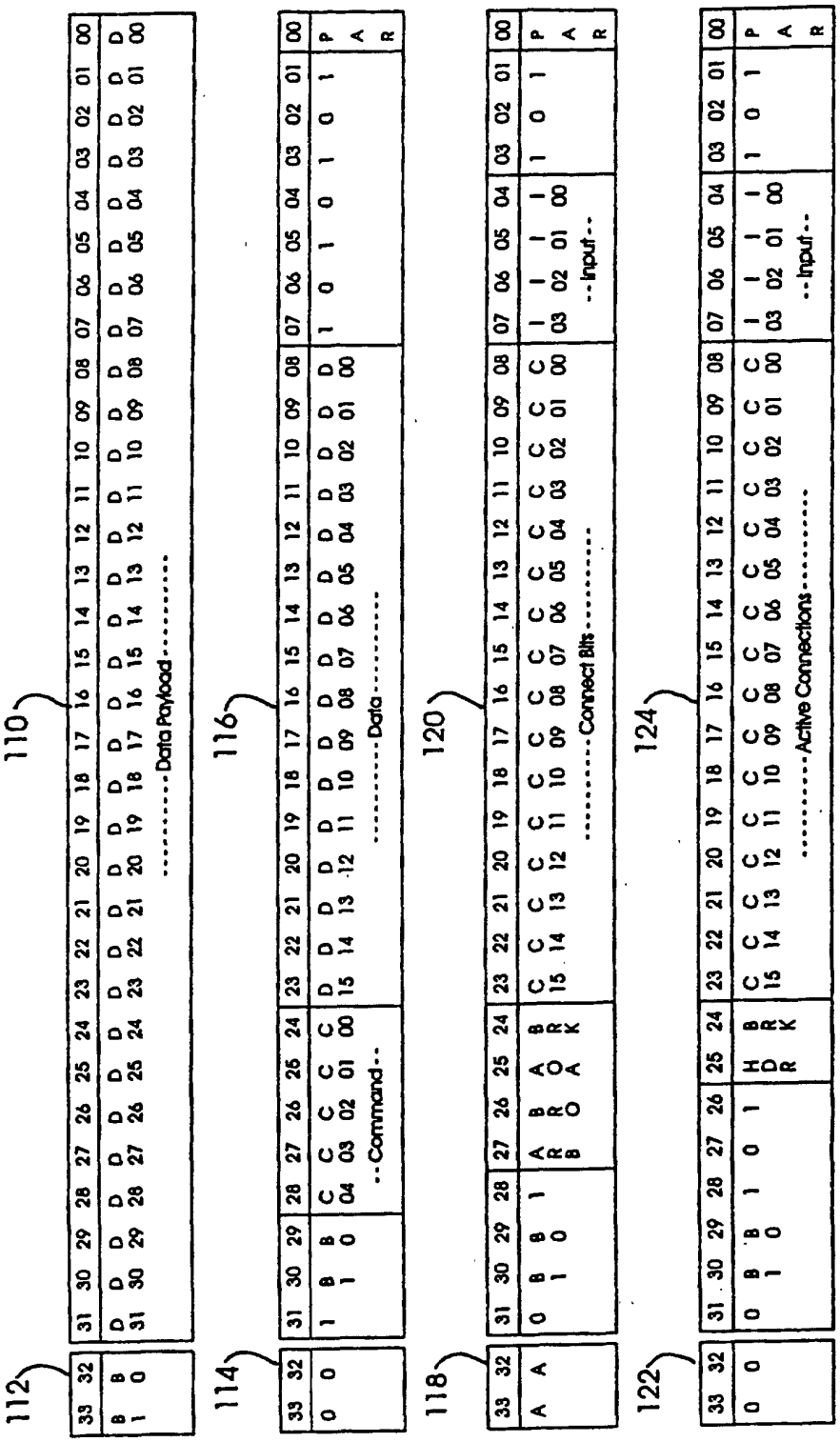

110
112
114
116
118
120
122
124
Data Payload
Command
Data
Connect Bits
Input
Active Connections
Input

FIG. 7

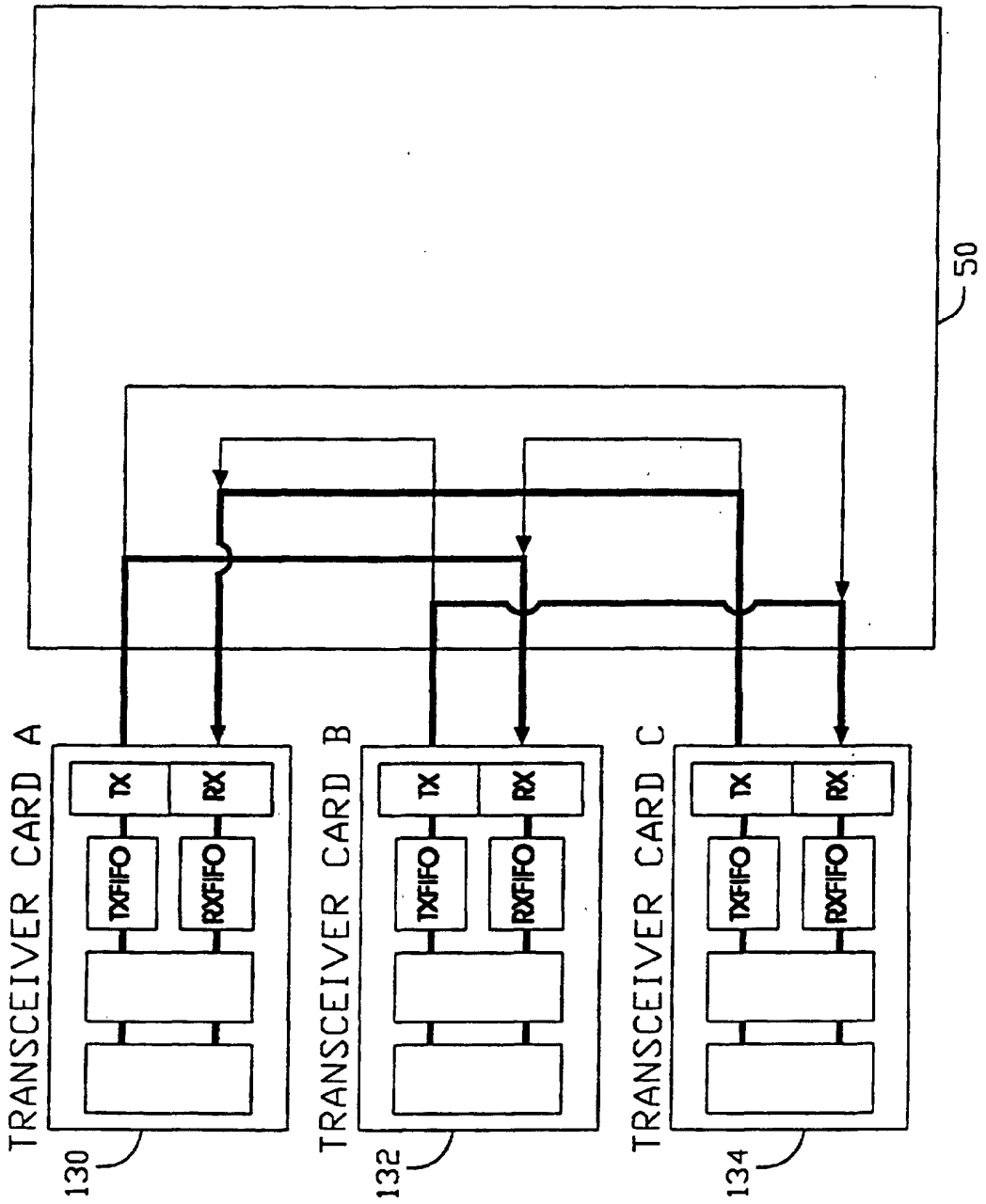

50
TRANSCEIVER CARD A
TX
RX
TXFIFO
RXFIFO
130
TRANSCEIVER CARD B
TX
RX
TXFIFO
RXFIFO
132
TRANSCEIVER CARD C
TX
RX
TXFIFO
RXFIFO
134

FIG. 8